6. Raj

Program Name: Raj.java Input File: raj.dat

Raj is trying to build a better calculator. Instead of a typical calculator which reads in-order expressions and evaluates them, Raj wants to build a calculator that takes in numbers and *assigns* the operations to yield the highest value expression. The current operations on Raj's calculator are pretty simple; he can only assign pairs of parentheses, and the operators for addition, subtraction, multiplication, and division. His calculator **can rearrange the numbers** if it helps him find the optimal value.

For example, the list **`4.0 5.0 3.0 2.0 1.0`** using Raj's new calculator should find the optimal expression yielding the maximum value to be:

`4.0 * 5.0 * 3.0 * (2.0 + 1.0) = 180`

Please help Raj design his calculator!

Input: The first integer N will represent the N data sets to follow. Each line will be a list of **n** floating point values.

Output: Each data set should return a single floating point value rounded to a precision of two decimal places representing the highest value expression that Raj could create by inserting the symbols **`( ) + - * /`** into the given expression.

Assumptions: Each decimal will fit into a Java double, and there will be at least one number in each of Raj's lists. **`Infinity`** is a valid largest value and output.

Sample Input:

```
3
4.0 5.0 3.0 2.0 1.0
1.0 1.0 1.0 1.0 -1.0
-1000.33 0.33 0.45 10.1
```

Sample Output:

```
180.00
5.00
30629.93
```